

# ソフトウェア概論 A/B

## -- データ構造 (4) --

(メモリモデル)

数学科 栗野 俊一 / 渡辺 俊一

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます

2020/11/20 ソフトウェア概

# 伝言

---

- 出席パスワード : 20201120
- 色々なお知らせについて
  - 栗野の Web Page に注意する事  
<http://edu-gw2.math.cst.nihon-u.ac.jp/~kurino>
- やる気のある方へ
  - 今日の資料は、すでに上っています
    - ▶ どんどん、先に進んでかまいません

# 前回の内容

---

ソフトウェア概論 A/B (2020/11/20)

## 前回の内容

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます

# 前回の内容

---

## □ 前回の内容：データ構造 (3)

### ○ 講義

- ▶ 多次元配列：配列の配列が作れる

### ○ 演習

- ▶ 複雑なデータ構造

# 本日(2020/11/20)の予定

---

ソフトウェア概論 A/B (2020/11/20)

# 本日(2020/11/20)の予定

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます

# 本日(2020/11/20)の予定

---

## □ 本日(2020/11/20)の予定

### ○ データ構造 (4)

▶ メモリモデル

## □ 本日の目標

### ○ 演習

▶ 課題の提出

# 課題

---

ソフトウェア概論 A/B (2020/11/20)

## 課題

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます

# 今週 (2020/11/20) の課題

---

## □ 今週 (2020/11/20) の課題

### ○ 課題 20201120-01: 文字列の入力

▷ ファイル名 : 20201120-01-YYYY.c (YYYY は学生番号)

▷ 内容 : キーボードから一行の文字列を入力し、その中にある 'a' の個数を入力する

## □ ※

○ ファイル形式は、いずれもテキストファイル(C 言語プログラムファイル)



# 先週 (2020/11/13) の課題

---

## □ 先週 (2020/11/13) の課題

### ○ 課題 20201113-01:

- ▶ ファイル名 : 20201113-01-QQQQ.c (QQQQ は学生番号)
- ▶ 内容 : 配列内の浮動小数点数の合計を求める Sum 関数

### ○ 課題 20201113-02:

- ▶ ファイル名 : 20201113-02-QQQQ.c (QQQQ は学生番号)
- ▶ 内容 : 文字配列に入った文字列の途中に文字を挿入する

### ○ 課題 20201113-03:

- ▶ ファイル名 : 20201113-03-QQQQ.c (QQQQ は学生番号)
- ▶ 内容 : 一行分の文字列を入力して、その中の文字列を全て大文字に変換する

### ○ ※ ファイル形式は、いずれもテキストファイル(C 言語プログラムファイル)

# 文字列の入力

---

ソフトウェア概論 A/B (2020/11/20)

## 文字列の入力

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます

# 文字列の入力

---

## □ 文字列の入力

- 「文字列」は、「文字」の配列(の途中に **EOS** が入ったもの)
- 「文字列を入力する」とは ? : 複数の文字を入力して配列に収める

## □ 文字列の入力方法

- 方法 1 : **scanf** の「%s」を使う -> 危険なのでやってはいけない
- 方法 2 : **gets** を使う -> 今では使えなくなった (やっぱり危険だから)
  - ▶ 方法 1 / 方法 2 は、「バッファオーバーフロー」の根源
  - ▶ 結果的にセキュリティホール
- 方法 3 : **fgets** を使う -> 安全な方法(推奨)

# メモリモデル

---

ソフトウェア概論 A/B (2020/11/20)

## メモリモデル

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます

# メモリのイメージ

## □メモリ

### ○セルの並んだ物

- ▶セルのサイズは 1 byte

### ○個々のセルには、別々のアドレス(番地)がついている

- ▶アドレスは 0..0 ~ F..F (16 進)
- ▶アドレスは、「セルの名前」として働く(アドレスが同じなら同じセル)

### ○セルの機能 (変数と同じ)

- ▶情報を記録できる
- ▶情報を取り出せる

番地	セル	コメント
0		番地は 0 から開始 / 値は 1 byte
1		
2		
⋮	⋮	
100	1	100 番地に 1 という値が入っている
101	10	101 番地に 10 という値が入っている
⋮	⋮	
F...FFF		最後は 16 進数で F..FFF となる

# メモリ・モデル

---

## □メモリ・モデル

- C 言語の変数のモデルの一つで、「変数をメモリセルの組み合わせ」として理解する

- ▶C 言語の「変数の振舞い」を「考えるための仕組み(モデル)」

- ▶!! 「『何か』のモデル」とは『何か』を理解するために利用可能な、「より簡単な仕組み」の事

- ▶!! 「C 言語の変数」を「メモリモデル」を通じて理解する/ 簡単な理解しやすい

## □実は..

- 多くの場合、「C 言語の変数」は実際に「メモリセルの組み合わせ」になっている

- ▶変数の性質(代入)はメモリの性質(記憶能力)から説明できる

## □char 型変数とメモリモデル ( sample-005.c )

- char 型変数は、一つのメモリセルだと考える事ができる

- ▶char 型変数は address を持つ

- char 型変数をメモリセルと同様に扱う事ができる

# メモリモデルと配列

---

ソフトウェア概論 A/B (2020/11/20)

## メモリモデルと配列

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます

# メモリモデルと配列

---

## □ 文字列とメモリモデル ( sample-006.c )

### ○ 文字列は、文字の並び

▶ 文字は char 型変数で記録できるので、文字列は char 型変数の並び

## □ 文字変数の並びと文字列 ( sample-007.c )

○ アドレスが判れば、変数の内容をアドレス経由で操作できる

## □ 配列宣言 ( sample-008.c )

### ○ 配列とは

▶ 「複数の変数の並び」の事 (個々の変数を「配列の要素」と呼ぶ)

### ○ 一次元の配列宣言 ( sample-008.c )

▶ 変数と同様に型名の後ろに「配列名[サイズ]」の形で宣言

▶ 「サイズ」の個数だけの変数が宣言される。

▶ 配列の要素は「配列名[0] ~ 配列名[サイズ-1]」という「名前」になる

▶ 添字 : 「[」と「]」の間には整数値が指定でき、配列の何番目の要素かを表す



# 文字列と文字型の一次元配列の関係

---

ソフトウェア概論 A/B (2020/11/20)

## 文字列と文字型の一次元配列の関係

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます

# 文字列と文字型の一次元配列の関係

---

## □ 文字列と文字型の一次元配列の関係

### ○ C 言語の文字列

▶ 文字の並んだ物 ( 文字コードが連続に記録されている )

### ○ C 言語の文字型変数

▶ 文字コードを一つだけ記憶できる

### ○ C 言語の文字型の一次元配列

▶ 複数の文字型変数が並んだもの

### ○ C 言語の文字列型の一次元の配列で文字列を記憶することができる

## □ 文字列の一部の操作方法 (sample-009.c)

### ○ 文字配列の要素を変更すればよい

## □ 文字列を利用した文字配列の初期化 (sample-010.c)

### ○ 文字配列の要素を文字列を利用して初期化できる

おしまい

---

ソフトウェア概論 A/B (2020/11/20)

おしまい

講義内容の静止画・動画での撮影、及び SNS 等への転載を固く禁じます